Resequencing of the Structural Stiffness Matrix to Improve Computational Efficiency

R. Levy

Telecommunications Division

Improved computational efficiency and reduction in core storage requirements in large-capacity structural-analysis computer programs are achieved by taking advantage of the typical sparseness in the stiffness matrix. In many cases, the favorable effects can be enhanced by resequencing the interconnecting terms of the stiffness matrix. This produces a relatively large empty region that can be bypassed and a relatively small, compact region that is moved into the core for computations.

In the past, the goal of resequencing has been to minimize the stiffness matrix bandwidth. A favorable alternative resequencing procedure that has the goal of reducing the matrix wavefront is described in this article. Comparisons are supplied to show the relative compactness that can be achieved for practical structural models with wavefront sequencing and with bandwidth sequencing. Examples show that wavefront sequencing can produce a savings in the time required for subsequent decomposition of the stiffness matrix.

Introduction

Conventional finite-element analysis of structures is typically implemented by any of the several currently available computer program packages that perform matrix interpretive analysis. When the structure to be analyzed entails many degrees of freedom, it becomes important to develop the analytical model to fit efficiently within the computer program capacity and to avoid excessive computation time for processing. Ground-based radar antenna structures in the 26-m and larger classes are practical examples from a wide range of structures that contain 1000 or more degrees of freedom for which judicious modeling can provide substantial computational benefits.

Large-capacity computer programs for structural analysis operate most efficiently when the computational procedure is formulated to capitalize on the typical sparseness of the structural stiffness matrix. Sparseness is

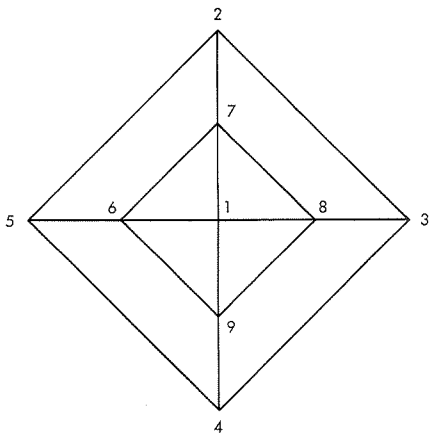
used to advantage by confining the computational operations to a compact region densely populated with non-zero coefficients and by omitting operations for the empty region in which the coefficients are zero. Consequently, it is often desirable to reorder a given stiffness matrix to make the region for computations as compact as possible and to make the empty region as large as possible. Traditionally, the compactness of the stiffness matrix has been measured by the matrix bandwidth, but an apparently more efficient equation-solving approach has been formulated to perform the computations within the region delineated by the matrix wavefront (Reference 1). The wavefront approach is attractive because the maximum wavefront is often considerably less than the maximum bandwidth, and it can never be greater.

The purpose of resequencing for an analytical model of a particular structure is to minimize the number of degrees of freedom that determine the maximum of the bandwidth or the wavefront of the stiffness matrix. Frequently, the number of degrees of freedom per node is approximately the same for all nodes. Therefore, it is reasonable to perform resequencing on the basis of the nodes and nodal connectivity, which is considerably simpler than resequencing on the basis of degrees of freedom and stiffness matrix connectivity. Algorithms of Rosen (Reference 2), Akyuz and Utku (Reference 3), and Cuthill and McKee (Reference 4) were designed to reorder nodes for bandwidth reduction. An algorithm to be described here has been found effective for wavefront reduction. A recent independent development by King (Reference 5), although described differently, appears to be related to the present method.

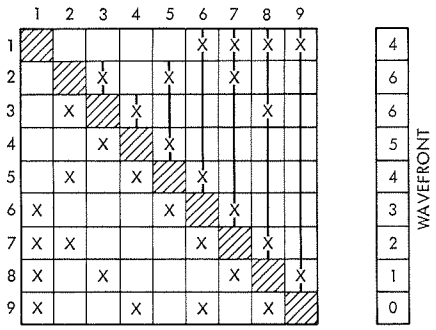
Resequencing procedures, either for bandwidth or wavefront reduction, do not ensure achievement of the optimal sequence. Results could be considered acceptable if there is an improvement from an initial sequencing in a local optimum sense, rather than convergence to an absolute minimum in a global sense. For this reason, it is sometimes useful to perform resequencing several times, with each cycle starting from a different initial sequence. This provides the opportunity to select the most favorable of the several local minimums that will be developed from the various cycles.

Wavefront Counting

The wavefront at a row of the nodal connectivity matrix is the number of active columns that follow the diagonal element; a column becomes active at the row containing the first entry for that column, and it remains active until that column is absorbed into the diagonal of the connectivity matrix. As shown in Figure 1, wavefront counting is accomplished by drawing heavy vertical lines for each column, beginning at the row at which the column becomes active and ending at the diagonal. To obtain the wavefront at each row, it is necessary only to count the heavy lines crossing each row.



(a) EXAMPLE STRUCTURE



(b) NODAL CONNECTIVITY MATRIX

Figure 1. Wavefront counting example

Alternatively, the wavefront at any row can be computed as equal to the wavefront at the preceding row, plus the number of new columns that become active. If the column corresponding to the row is currently in the wavefront, then the wavefront is reduced by one. The subtraction is made to account for columns that leave the wavefront by virtue of reaching the diagonal. This method of wavefront counting is explained for the rows of Figure 1a as follows:

Row 1: The wavefront is 4, because columns 6–9 become active at this row and the wavefront at the non-existent preceding row is defined to be 0.

Row 2: Columns 3 and 5 become active; therefore, the wavefront is $2 + 4 = 6$.

Row 3: Column 4 is added to the active list; because Column 3 was in the wavefront, 1 is subtracted. Therefore, the wavefront is $6 + 1 - 1 = 6$.

Rows 4–9: All columns have been activated (matrix has filled), and no new columns can enter the wavefront. Therefore, at each row 1 is subtracted from the preceding wavefront, since one column always reaches the diagonal and leaves the wavefront.

The isolated column to the right of the connectivity matrix in Figure 1b gives the wavefront at the row. The maximum wavefront is 6, which occurs at both the second and third rows. It will be shown that this can be reduced by resequencing.

Wavefront Sequencing

A resequencing approach that is recommended here for its conceptual simplicity can be called a "minimum growth" method. That is, assuming that resequencing has already been used to identify the nodes that will constitute the new first i nodes, the node to be selected as the new $(i + 1)$th node is the one for which the wavefront at the $(i + 1)$th node will represent the smallest increase with respect to the wavefront at the ith node. In accordance with the alternative method just described for wavefront counting, the increase can be a positive integer, zero, or negative unity. Very often, any one of several nodes will give the same minimum wavefront increase for the $(i + 1)$th node, and, when this happens, the first node encountered is chosen. This tends to make the new sequencing as close as possible to the original. The question remains, of course, as to how to choose the node to be first in the new sequence. The choice could be based upon one of the following alternatives:

(1) Choose the node that will give the minimum wavefront for the first row. This is equivalent to choosing the node with the minimum number of connections. Break ties by picking the first node encountered.

(2) When the connectivity matrix involves n nodes, perform resequencing n times, picking each node in turn to be the first. At the end, select the most favorable of the n sets of resequences. This method is feasible if the number of nodes is not excessive.

(3) Perform resequencing k (where $k \leq n$) times. Each time, pick at random the node that is to be first. This procedure is suggested if n is very large; nevertheless, more than one cycle of resequencing should be tested for possible improvements.

(4) Assign one particular node to be the first in accordance with prior inspection and reasoning.

The resequencing operations can be conveniently organized in a tableau format for manual execution. This format employs the original nodal connectivity matrix and a new table listing the change in wavefront for each possible selection of the next node to be placed in the new

sequence. Tableau resequencing will be demonstrated for the structure of Figure 1. Figure 2 shows the progressive changes (heavy lines) in the combined tableau as each next node is selected to be placed within the new sequence. For clarity, the figure is subdivided into the stages associated with selecting one successively ascending node at a time. The general procedure that is employed repetitively for selection of the successive nodes is described below.

Wavefront Change Table Operations for Figure 2

Step 1: Enter the number of the new node that is about to be selected in the new sequence heading list.

Step 2: Count the number of new connections at each row. This is equal to the sum of the entries in the connectivity matrix that do not occur at columns already in the wavefront. The columns that are in the wavefront are identified by vertical lines previously drawn through these columns. The change in wavefront at each row is the sum of the new connections. The change is reduced by one if the row is already in the wavefront. Enter the change in wavefront in the column of the node to be selected and opposite the associated row.

Step 3: Select the smallest change in wavefront from the column of wavefront changes (first occurrence in case of ties). Circle the row entry and draw a horizontal line across the wavefront change table. This is the new row to be placed in the sequence. Add the wavefront change to the preceding wavefront and enter the value at the space provided at the bottom of the table.

Connectivity Matrix Operations for Figure 2

Step 1: Strike the column of the node just selected (a double line used in Figure 2). Enter the new sequence number at the bottom of this column in the space provided.

Step 2: Strike the row of the node just selected.

Step 3: Strike all unstruck columns that have connections in this row.

Additional Notes on Figure 2

Stage 1: The first node is selected with the objective of minimum wavefront in the first row of the new sequence. Original node 2 is the first one encountered with the minimum number of 3 connections. Original node 2 becomes the new first node, the wavefront at the new first row of the connectivity matrix is 3, and columns 3, 5, and 7 of the connectivity matrix become activated.

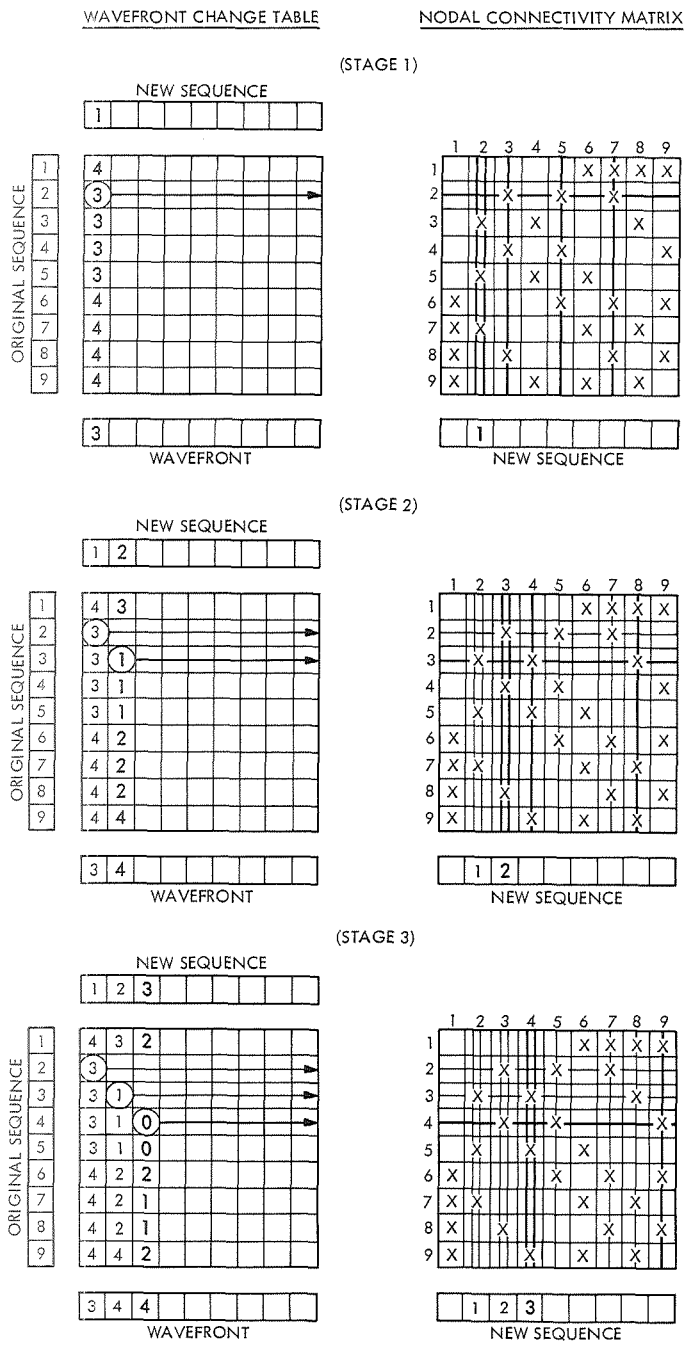


Figure 2. Tableau resequencing example

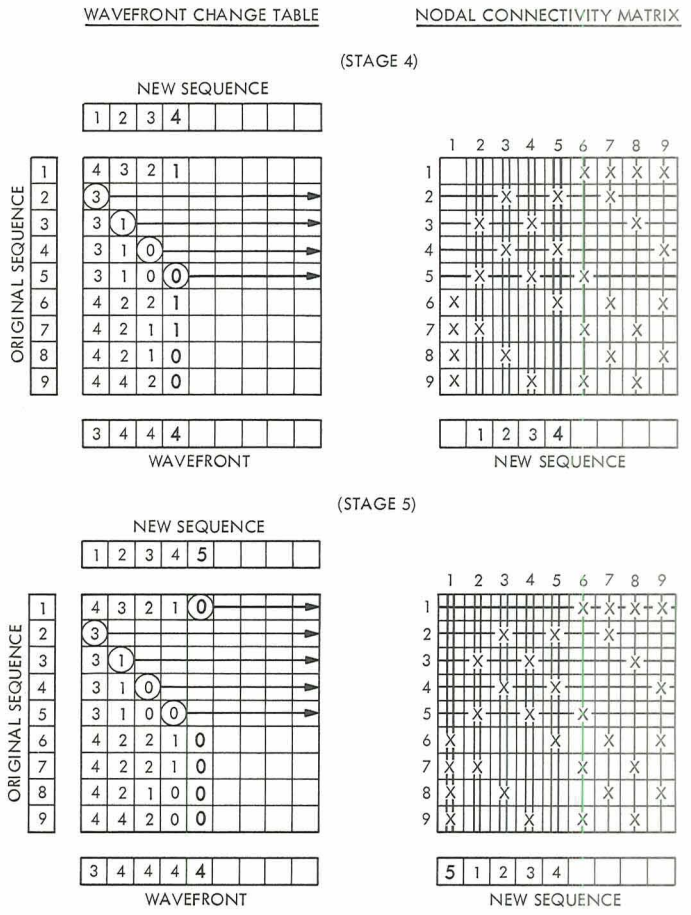


Figure 2 (contd)

Stage 2: Original node 3 causes columns 4 and 8 to become activated, but node 3 has been activated previously; thus, the net increase in wavefront is only one. Original node 3 becomes the new second node. Columns 4 and 8 are added to the active list, and column 3 leaves the wavefront.

Stage 3: Original node 4 activates column 9, but there is no net increase in wavefront because column 4 was in the wavefront. Original node 4 becomes the new third node, and column 4 leaves the wavefront.

Stage 4: Original node 5 activates column 6, but there is no net increase in wavefront because column 5 was in the wavefront. Original node 5 becomes the new fourth node, and column 5 leaves the wavefront.

Stage 5: Original node 1 does not activate any additional columns, but there is no net decrease in wavefront because column 1 was not on the active list. Original node 1 becomes the new fifth node.

At the completion of stage 5, there are no remaining columns to be activated, and the wavefront has filled the connectivity matrix. The remaining nodes can be added to the new sequence in arbitrary order, and the wavefront will decrease by unity at each successive stage.

Results and Conclusions

Tableau resequencing can be performed without difficulty for matrices with orders of 20 to 30. When the order becomes much larger, the tableau becomes cumbersome, and it is advisable to perform resequencing by computer. A program has been written to automate the procedure by performing essentially the same operations as those for executing the tableau. The largest model for which the program has been used to date has a connectivity matrix with over 500 nodes. In one test of the program for this model, the initial sequencing was developed by inspection and study of the connectivity. The maximum wavefront was 43. One resequencing computer cycle reduced the initial maximum wavefront to 30. In another test, the initial sequencing was scrambled at random, which produced a maximum wavefront of 92. One computer cycle reduced this to 40. As the result of six additional computer cycles, the maximum wavefront was reduced to 27; in these, the starting node for each cycle was chosen at random.

On the UNIVAC 1108 Exec-8 computer, the computation time for this structure averaged about 15 s/cycle. For a structure with 75 nodes, the program averaged more than 2 cycles/s. The computation time per cycle varies considerably, depending upon how quickly the connectivity matrix fills or upon the number of nodes processed before it can be determined that the current cycle is to be aborted because there will be no reduction of an existing maximum wavefront.

Table 1 shows some additional comparisons for resequencing tests to reduce either bandwidth or wavefront. These tests were performed for

Table 1. Effects of resequencing on bandwidth and wavefront[a]

Component type	Nodes	Initial size		Size after resequencing	
		Bandwidth	Wavefront	Bandwidth	Wavefront
Antenna pedestal	56	40	20	23	11
Quadripod	66	41	18	16	8
Coarse grid reflector	83	73	30	24	12
Azimuth–elevation reflector	466	200	52	75	29
Polar reflector	506	—	92	—	28

[a]Bandwidth sequencing method described in Reference 2; wavefront sequencing method described in this article.

the analytical models of structural components used on ground-based radar antenna systems. The table shows that, after sequencing, the wavefront tends to be less than half of the bandwidth, which can provide definite computational advantages in subsequent analysis. The savings in computation time depends to some extent on the type of analytical problem to be solved and the efficiency of the approach used to process the stiffness matrices. For statics problems, comparisons have shown that about 20% savings in stiffness matrix decomposition time could be achieved by using wavefront sequencing instead of bandwidth sequencing. In eigenvalue problems, the possible savings appears to be considerably greater. Table 2 shows a few cases from Table 1 where the comparable decomposition times were available for both types of sequencing.

Table 2. Stiffness matrix decomposition times

Component type	Problem type	Decomposition time, s	
		Bandwidth sequencing	Wavefront sequencing
Coarse grid reflector	Statics	10.4	7.5
Azimuth–elevation reflector	Statics	260.1	215.2
	Eigenvalue	3604.5	635.4

The experience with the computer program indicates that minimum growth sequencing is effective, rapid, and capable of producing worthwhile economies. The savings occurs in the reduction or elimination of lengthy effort to study a particular structural model for the purpose of generating an acceptable sequencing pattern, as well as in the reduction of computation time as the result of processing a relatively compact stiffness matrix.

References

1. Melosh, R. J., and Bamford, R. M., "Efficient Solution of Load-Deflection Equations," *J. Struct. Div., Proc. ASCE*, Vol. 95, No. ST4, pp. 661–676, Apr. 1969.
2. Rosen, R., "Matrix Bandwidth Minimization," *Proceedings of the 23rd National Conference of the ACM*, p. 585. Brandon/Systems Press, Inc., New Jersey, 1968.
3. Akyuz, F. A., and Utku, S., "An Automatic Node-Relabeling Scheme For Bandwidth Minimization of Stiffness Matrices," *AIAA J.*, Vol. 6, No. 4, pp. 728–730, Apr. 1968.
4. Cuthill, E., and McKee, S., *Reducing the Bandwidth of Sparse Symmetric Matrices,* Applied Mathematics Laboratory Technical Note

References (contd)

AML-40-69. Naval Ship Research and Development Center, Washington, D.C., June 1969.

5. King, I. R., "An Automatic Reordering Scheme For Simultaneous Equations Derived From Network Systems," *Int. J. Numer. Meth. Eng.*, Vol. 2, pp. 523–533, 1970.